40 Years of Oracle Forms:

Time for Modernization or Migration?
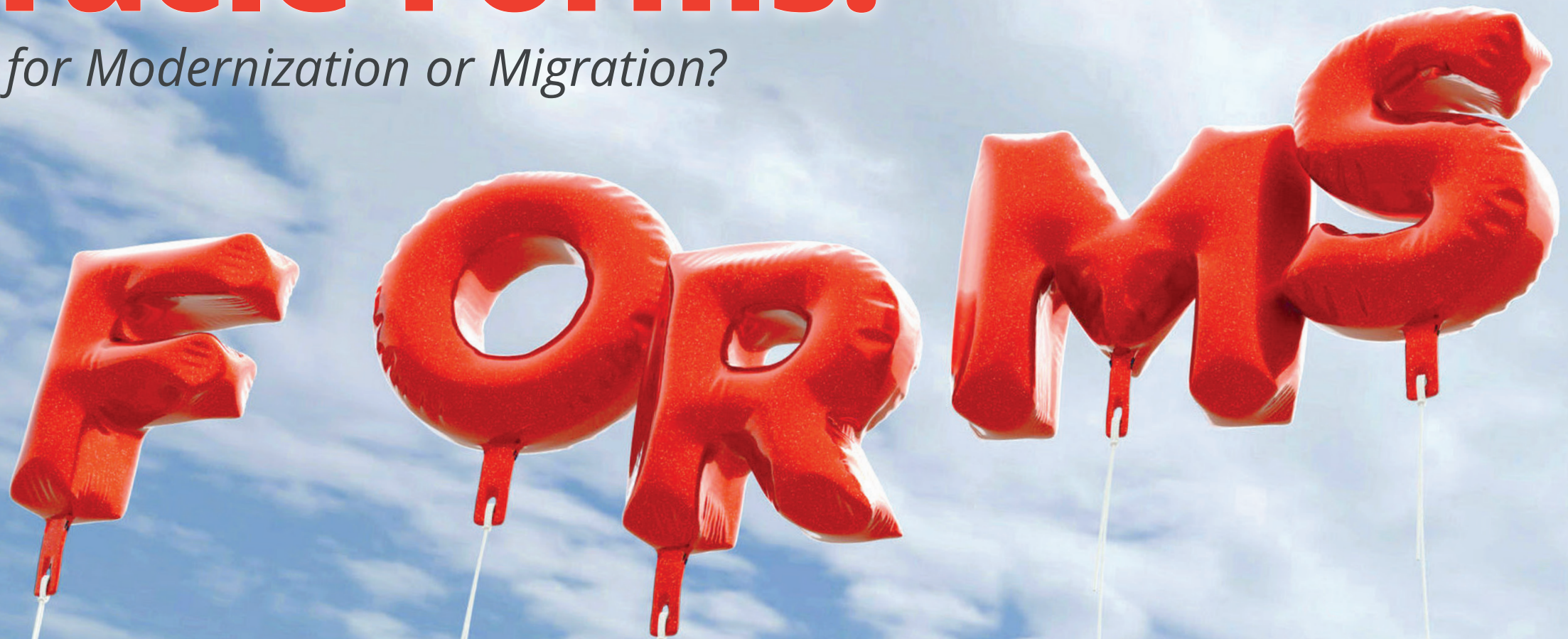
- **Are You Coworking Yet?**
- **No Longer Allowed: RAC on SE2**
- **Code You Should Never See in PL/SQL – Part 4**

EOUC

EMEA
ORACLE
USERGROUP
COMMUNITY

Are You Coworking Yet?

page 6

Code You Should Never See in PL/SQL – Part 4

page 16

40 Years of Oracle Forms

page 22

Have a comment? Suggestion? Want to submit an article proposal? **Here's how to do it.**

Editorial

Dear Oracle User Group Community Members

What a pleasure that you have chosen to explore this issue of ORAWORLD.

In this edition you will find some interesting facts about coworking. I really enjoyed reading the article on **page 6**. Practice says that ideas coming through inspiration and with a help of other people. Learn more about the history and search for community-based resource in your country. I found several nice places in mine.

ORAWORLD is a diverse electronic magazine and in this issue, you also can find an interesting platform for electronic music. The creator of this project, **Sam Aaron**, was voted best speaker at the Riga DevDays 2019 conference on which I participated as organizer. He is really a cool speaker and his music platform is fun to use, easy to create, share with others, and it is free to use.

This issue is covering a lot of Oracle technologies like Oracle RAC with Standard Edition 2. Please, explore the article from Andrew Lacy, and find out why RAC SE2 is not supported on the latest database edition. You can also find the final part of the series about mistakes you should avoid in PL/SQL, by Steven Feuerstein.

This magazine celebrates 40 years with Oracle Forms. I found this article by Frank Hoffmann very interesting. For myself I understood how Oracle Database and Forms and Reports became so popular. I should admit we are still using Forms and Reports in our company.

Andrejs Vorobjovs
LVOUG president (Latvia), EOUC

In June, there was the traditional EOUC leader summit. Heli from Finland has made a short resume about the meeting. On behalf of the organisers, I hosted this meeting in Riga, Latvia. The meeting was very valuable for all Oracle User Groups, and we decided to continue this tradition on our own.

I hope you enjoy browsing through this issue! Please remember to submit your content for the upcoming issue online on our website: www.ORAWORLD.org.

Yours, Andrejs Vorobjovs
LVOUG president (Latvia), EOUC

Editorial

Giveaway

www.commitstrip.com

Work & Life

Are You Coworking Yet?

Christian Luda

We are living in a time of change. As technology keeps evolving, work is changing. Old jobs disappear, while new jobs are being created. But not just the jobs themselves change, the way people work does as well. Cloud Computing and the ubiquity of WiFi are two main factors for the rise of new work forms like the phenomenon of "coworking". But what exactly is coworking and how did it start?

The concept of coworking is based on a simple idea: building a community and offering space for independent professionals and people who have workplace flexibility; and the assumption that people work better together than isolated at home. These people still work independently and for different companies, but they all share the same values and can profit from possible synergies that happen at coworking spaces.

While coworking always needs a space offering the required work environment (e.g. seats, WiFi, printers etc.) the community aspect is just as important. That's why models like business incubators or executive suites are not considered coworking as they lack the social and collaborative aspects.

Freelancers, digital nomads, start-up entrepreneurs and their staff make up the majority; however, the number of large company employees that are coworking is increasing. Tech giants such as Microsoft, Facebook, IBM, and Oracle let some of their employees use coworking spaces hoping to get new ideas and inspiration through the interaction with the innovative startup scene; the possibility of discovering new talent is another factor.

Today, coworking spaces can be found all over the world but – like so many other things involving tech – it all started in Northern California. In 2005, **Brad Neuberg**, a computer scientist unhappy with his job at a start-up, comes up with the idea to combine the structure and community of a company with the freedom and independence of a freelancer.

Soon after, Neuberg opens the "San Francisco Coworking Space" at the "Spiral Muse", a location owned by a feminist collective in San Francisco's Mission district. A friend of his who is part of the collective has offered him to rent their space for two days a week.

This space will later be widely acknowledged as being the first

coworking space in today's sense. It should be noted though that there had been other experiments with elements of what we understand as coworking at different places in the world before. For example **c-base**, one of the world's first hackerspaces founded in Berlin in 1995, made WiFi networks available in 2002 and promoted free internet access for the public. That same year, the "Schraubenfabrik" opened in Vienna as a "community center for entrepreneurs".

While Neuberg coins the word coworking for his idea, when looking for a domain name for a coworking wiki he finds out someone else has already come up with the same name, but for something totally different: Bernie de Koven who then owns the domain coworking.com had built the Coworking Institute, an organization aiming to raise awareness for collaborative work technologies. When they find out about each other Neuberg and de Koven briefly connect on the phone and laugh about this coincidence.

Meanwhile, the "San Francisco Coworking Space" takes off slowly. After a first month with no visitors at all, Neuberg does not give up on his idea and starts promoting his space more intensely thus attracting a growing number of coworkers. When the "Spiral Muse" closes about a year later, Neuberg along with Tara Hunt and Chris Messina (who would later invent the hashtag) soon open new a space called "the Hat Factory" – the first full-time coworking space – and that's when the popularity of coworking really starts to rise.

In 2006, the famous "Citizen Space" opens in San Francisco while two New Yorkers, Amit Gupta and Luke Crawford, start the coworking group "Jelly" receiving worldwide media coverage and spawning more groups in other cities. The same year, New York City's first coworking space gets started in Williamsburg, Brooklyn. "The Change You Want to See Gallery"

– later re-named **"Brooklyn Collab Studio"** and relocated to the Greenpoint neighborhood – still exists today making it the world's longest running coworking space.

By 2008, most major cities in Northern America and Europe have developed their own coworking communities. At that time, traditional media really takes notice; the New York Times publishes its **"They're Working on Their Own, Just Side by Side" article** portraying Brad Neuberg. 2008 also sees the introduction of the Coworking Visa program helping travelling coworkers to work at other spaces around the world. The following year, the first Global Coworking Unconference Conference takes place, while Todd Sundsted, Tony Bacigalupo, and Drew Jones release "I'm Outta Here" – the first book on coworking.

Between 2006 and 2015 the number of available seats at coworking spaces doubles each year – a trend that shows no signs of letting up.

Today, coworking has become a big business itself with companies specializing in providing shared workspaces and services. **WeWork** is the most prominent one: the New York City-based company boasts a valuation of 40 billion US dollars and has a total number of 551 spaces in 97 cities in 36 countries around the world. WeWork's 268,000 members get access to health insurance, the company's social network, workshops and other benefits.

Another big player is **Spaces** from Amsterdam. Its members (including Oracle employees) have worldwide access to the company's business clubs, networking lunches, and speaker events. The services include reception and mail handling.

How big coworking has become is also exemplified by the various conferences dedicated to it in the EMEA alone: This year already saw the **Coworking Konferenz & Barcamp** in Mannheim, Germany in March, the **Coworking Spain Conference** in April, and the **Coworking Conference Nigeria** in July, plus there will be the **Salon Coworking** at Stade de France, Paris on October 17 as well as the **Coworking Europe** in Warsaw from November 13 - 15.

If you have become interested in trying out coworking yourself or want to learn more about the subject, you will find plenty of useful information on the **Coworking Wiki**, a free and community-based resource for coworkers, space owners or just coworking enthusiasts. The site offers a lot of links to blogs and media coverage, marketplace directories, groups and collectives, academic research, tools, and much more.

Of course, if you are living in a bigger city, a quick Google search should help you find all the available spaces for your region so you can start coworking just in time to join the festivities when this interesting work form will celebrate its 15th anniversary in 2020!

Sources
http://codinginparadise.org/ebooks/html/blog/start_of_coworking.html
http://wiki.coworking.com/w/page/16583831/FrontPage
https://en.wikipedia.org/wiki/Coworking
http://www.deskmag.com/en/the-history-of-coworking-spaces-in-a-timeline
https://expandedramblings.com/index.php/wework-statistics

Christian Luda

Number of the Month: 20,000

"Good neighbor" Google wants to create 20,000 new homes in the San Francisco Bay Area

Techs & Nerds

In June, Google announced to give no less than a billion US dollars (about 890 million euros) to contain the housing crisis that's plaguing San Francisco and the surrounding Bay Area. The sharp rise in real estate prices and the cost of living in the region is a problem for which Google and other tech companies are held significantly responsible for. Not only do they attract thousands of well-paid developers, they also buy a lot of buildings and land to expand their headquarters, thus driving up the prices.

An average house in Silicon Valley costs more than a million dollars – the highest value in the United States. The market has become so much out of control that even tech workers have troubles to buy a home. According to a **2018 Blind survey** that asked employees of 13 Bay Area tech companies (including Oracle) whether they could afford to buy a house in the region, 59 % of the participants (more than 100 of each company) said they couldn't.

According to a **survey by Edelman**, in 2018, 49 % of Bay Area residents stated that they were considering moving away from California because of the high cost of living.

The big tech companies have been under pressure to do something about the situation for quite some time. "As we work to build a more helpful Google, we know our responsibility to help starts at home. For us, that means being a good neighbor in the place where it all began over 20 years ago: the San Francisco Bay Area", declared Google CEO Sundar Pichai in a **blogpost** on June 18.

According to the post, Google has created a 10-year plan to repurpose a 750 million dollars-worth portion of its own land originally planned for office and commercial space as residential housing. This shall lead the way for 15,000 new homes for all income levels. In addition, an investment fund of another 250 million dollars is planned to provide incentives for housing developers which shall support the building of at least 5,000 affordable housing units.

Google is not the first tech giant recognizing the problem. In January, Microsoft had announced to invest 500 million US dollars to help create affordable housing and fight homelessness in the greater Seattle area.

There has been no news yet if Oracle is considering a similar initiative too. Larry Ellison however has no real estate problems. Oracle's co-founder and former CEO owns various homes in Silicon Valley, Lake Tahoe, and Japan, plus 98 % of a Hawaiian island. You can have a look at his impressive real estate portfolio at **this gallery compiled by Business Insider**.

After all, one could argue like **New York Times writer E. Tammy Kim did in January** that it's not really the companies' task to fix problem, but rather politicians in demand of coming up with solutions. Higher taxes for big companies could be used to fund affordable housing.

EXPLORE YOUR OPTIONS

No Longer Allowed: RAC on SE2
Andrew Lacy

From version 19.x of the Oracle database, Real Application Cluster (RAC) is no longer allowed with the Standard Edition 2. As a user of the Oracle Database Standard Edition 2, what other options do I have now?

Monday morning, but I'd already had my first coffee and a customer wanted to order a Standard Edition 2 database on an Oracle Database Appliance (ODA) running Real Application Cluster (RAC). So it was a nice start to the day.

Well, it would have been, except that I had read a twitter thread over the weekend, where the news came out that RAC with

Standard Edition 2 (SE2) would be no longer allowed in the future.

Before I rang the customer, I considered what this really meant for him, what his different options could be. The customer needs a database that is highly available. The standard high-availability solutions with the Oracle Database are Dataguard and RAC. Now both would only be available with the Enterprise Edition.

Techs & Nerds

What else can we do here? Install RAC with SE2 and stay on version 18? Look at different high availability solutions apart from Dataguard and RAC? Move to the Enterprise Edition? How could I help the customer make this decision?

RAC, SE2, Version 18

Of course, I can install a RAC with version 18 of the Standard Edition 2 Oracle Database. Then I have Premier Support for my highly available solution until May 2021 (MOS 742060.1). Of course I can still open a service request with Oracle after this date, but I will get no further security patches unless I pay for Extended Support. However, there is no Extended Support for version 12 or version 18 of the Oracle Database. This is because version 18 is really a version 12.2.0.2 and version 19 is really a version 12.2.0.3. So, version 19 is really the last "patch set" of version 12 and is therefore the only version that has extended support. These were the same rules for versions 11.2.0.3.0 and 11.2.0.4.0. RAC is difficult to install and complex to operate. For a database that is so important that we choose RAC as the high-availability (HA) solution, we are going to need security updates.

Pro	Know-how available in RAC	Very highly available	
Con	Difficult to install	Complex to operate	No security patches after May 2021
Licences & Hardware required	2 x Oracle Database SE2 v18	2 x operating system	2 x server

Single Instance

Of course with version 19, I can continue to use the single instance option on SE2. I have no high availability. But then, RAC is difficult to install correctly and complex to operate. This

means that we need skilled people to set it up and operate it, adding to our costs. Do I really need this level of high availability? The single life is so much easier ;-)

If I automate the Oracle installation and keep a spare machine, I can basically reduce the downtime to the time required to restore the backup. I can certainly reduce the data that I could lose, to less than 15 minutes. Is this level of downtime and data loss not acceptable?

Pro	Basic know-how for single instance	Security patches available after May 2021	Low cost (1 x SE2 licence)
Con	No high availability		
Licences & Hardware required	1 x Oracle Database SE2 v18	1 x operating system	1 x server (perhaps a 2nd server as fallback)

Standby (Dbvisit / Libelle)

Perhaps the database is large and needs a longer time to restore. Or losing e.g. 15 minutes of data is not acceptable. Dataguard, the standby solution from Oracle is only available with the Enterprise Edition. But there are external suppliers who offer a standby solution for the Standard Edition 2.

With RAC, there is a SPOF (Single Point of Failure) in the storage and also in the RAC system. With a standby solution, I have a second copy of the data and have real disaster recovery in this high-availability solution.

Pro	Standby easier than RAC, real disaster recovery	Security patches available after May 2021	Avoids higher cost of Enterprise Edition
Con	Switchover/failover is a manual process	Some data loss (seconds)	
Licences & Hardware required	2 x Oracle Database SE2 v18, 2 x Dbvisit licence pro database; with many databases per server, look at Libelle	2 x operating system	2 x server

Oracle Failsafe

This is a very interesting solution from Oracle for users running their Oracle databases on Windows. Based on Microsoft Windows cluster, Oracle Failsafe offers high availability for the application and the database. The data is stored on shared disks. A virtual IP is automatically moved from one Windows node in the cluster on server failure.

A further advantage to Failsafe as a possible HA solution is that I may only need one Standard Edition 2 licence, if I keep to the 10-day rule. This means if the sum of usage of the failover is less than 10 days, I only need to licence the primary. See the following URL for the exact explanation.
https://www.oracle.com/assets/data-recovery-licensing-070587.pdf

Pro	High availability from Oracle for the application and the database	Security patches available after May 2021	Avoids higher cost of Enterprise Edition. Only 1 x SE2 licence required (10 day rule)
Con	Unusual technology, less available know-how	Some data loss (seconds)	
Licences & Hardware required	1 x Oracle Database SE2 v18	2 x Windows Standard Edition	2 x server

Oracle VM

Using Oracle VM (OVM), I can reduce the number of cores for my Oracle Database Enterprise Edition, assigning only a set number of cores and pinning them to the VM. While OVM does not have the full functionality that is available in VMware, this licensing advantage can save a lot of money.

Pro	High availability from Oracle; can migrate VM from one physical machine to another	Security patches available after May 2021	Can reduce cost of licencing with EE
Con	Unusual technology, less available know-how	Almost zero downtime, zero data loss	
Licences & Hardware required	1 x Oracle Database EE/SE2 per VM	1 x operating system per VM	2 x physical server (for HA)

RAC One Node + EE

With RAC One Node, the database can only be running on one server at a time, so I can reduce the necessary licences. Whether I need one or two database licences here is decided by the 10-day rule (see URL above with Oracle Failsafe).

Particularly when combined with OVM, this is a very attractive proposition, reducing the number of Oracle Database licences required and yet offering a good HA solution.

Pro	Zero data loss	Security patches available after May 2021	Avoids higher cost of Enterprise Edition. Only 1 x EE + RAC One Node licence required (10 day rule)
Con	Unusual technology, less available know-how	Downtime (seconds to minutes) for switchover	
Licences & Hardware required	1 x Oracle Database EE + 1 x RAC One Node	2 x operating system	2 x server

RAC + EE

If I must have this level of high availability, I can of course still setup a RAC with the Enterprise Edition. The costs will be higher, but perhaps I can control those by reducing the number of cores in the choice of server hardware (or virtualising with OVM). And of course I have Dataguard "for free" with EE, so I can combine that with RAC to have a really good HA solution. Add in parallel queries, flashback and the many other options included in EE and although more expensive than SE2, this could be an attractive possibility.

Pro	All features of EE, same know-how as running RAC with SE2	Security patches available after May 2021	Zero data loss
Con	More expensive then RAC on SE2		
Licences & Hardware required	2 x Oracle Database EE, 2 x RAC option	2 x operating system	2 x server

Oracle Cloud

Oracle has a good offer for SE customers who want to move to the cloud. You can convert your SE Oracle RAC Databases into an Autonomous Database BYOL.

Conversion Ratios for Oracle Database Standard Edition, Oracle Database Standard, and Oracle Database Standard Edition 2 (Oracle Database Standard Edition Programs) are the following: For 1 - 8 OCPUs of either Oracle Autonomous Transaction Processing Cloud or Oracle Autonomous Data Warehouse Cloud you may activate up to 4 OCPUs of the BYOL Cloud Service for each supported Processor License of Oracle Database Standard Edition Programs (where a Processor is defined as equivalent to an occupied socket). You can find more info here: **http://www.oracle.com/us/corporate/contracts/paas-iaas-universal-credits-3940775.pdf**

Pro	You finally move your Oracle Database to the cloud. You have wanted to do this for ages! :-)	Security patches available after May 2021	Lower costs through BYOL (see details above)
Con	To avoid high latency, consider moving your application to the cloud. Also consider backup and single sign-on	With RAC, zero data loss	
Licences & Hardware required	RAC or RAC one-node subscription in the cloud	No operating system in own computer centre	Zero server in own computer centre

Wait SE, SE1, SE2? There already were some changes with the standard edition

The Premier Support for 12.1.0.1.0 ended early in August 2016 and the Premier Support for 11.2.0.4.0 was extended until December 2018. With version 12.1.0.2.0, one had to migrate from SE/SE1 to SE2. With SE2, we have maximum 8 threads per Instance with RAC (16 with a single instance). We also are now reduced to 2 Sockets on the motherboard (this was 4).

Summary...

Due to this licensing change with RAC on SE2 from version 19 onwards, the customer will have the following changes:

- Licence changes
- More costs
- Higher complexity through new technology
- More downtime

And which option did the customer choose? For now the customer wants a RAC on SE2 on ODA and currently plans to upgrade to the Enterprise Edition with version 19.

Andrew Lacy
Logicalis GmbH, Presales Consultant
andrew.lacy@logicalis.de

Techs & Nerds

Code You Should Never See in PL/SQL Part 4

Steven Feuerstein

Here is the 4th and last part of our series about mistakes you should avoid in PL/SQL!

Hiding errors

(suggested by Patrick Jolliffe, **@jolliffe**)

When an exception is raised in your code, you really should do something about it, namely:

- Write the error to your log table.
- Make nice with your users (don't show them an error stack or "internal" error message).
- Analyze the error and get it fixed ASAP.

So, what you should never do is something along these lines:

Bad Code

```
BEGIN
  your_code_here;
EXCEPTION
  /* This will completely ignore any error */
  WHEN OTHERS
  THEN
     NULL;
END;
/

BEGIN
  your_code_here;
EXCEPTION
  /* This will write information to the output buffer. Can you see that
     in production? I doubt it! */
  WHEN OTHERS
  THEN
     DBMS_OUTPUT.PUT_LINE (
        'OMG! Something went wrong: ' || SQLERRM);
END;
/
```

The problem with these exception handlers is that you are "swallowing up" errors – you will never see them, your users will never know something went wrong, bugs will persist, and your karma will take a major downturn. Just sayin'.

Cleaned Up

What should you do instead?

First of all, there are certainly situations in which an exception is raised and you actually, truly "don't care" – because it's not really an error. The classic example of this scenario is doing a SELECT-INTO that doesn't find a row. PL/SQL will raise the NO_DATA_FOUND exception, but that's not necessarily an error – it could just be a "data condition". As in: there's no row for that criteria, so now I will take branch A in my code instead of branch B.

So ignoring an exception is justified. But in that case, you should only ignore the single (or perhaps several) exception that is not really an exception, as in:

```
BEGIN
  BEGIN
     SELECT id
     INTO l_id
     FROM my_table
     WHERE nm = NAME_IN;
  EXCEPTION
     WHEN NO_DATA_FOUND
     THEN
        /* It's OK for the row not to exist. This just means we do A in-
stead of B */
        l_next := 'A';
  END;

  IF l_next = 'A'
  THEN
     NULL;
  ELSE
     NULL;
  END IF;
END;
/
```

And I do encourage you to include a comment explaining why you are not logging and re-raising the error.

More generally, I urge you to carefully consider whether or not an exception should be ignored.

If the exception is to be ignored, then ask yourself: do you want to know this happened? If not, then just be sure to "ignore" / handle specific named exceptions, never WHEN OTHERS.

If the error should not be ignored, then you will want to log it and continue.

I have several places in the backend code for the **Oracle Dev Gym** in which something went wrong (for example, an error occurred when trying to send an e-mail verification of some step taken), but I do not want to stop processing, I do not want to bother the user. So, I log the error, but allow execution to continue.

Here's the exception section for my send_mail procedure:

```
EXCEPTION
   WHEN OTHERS
   THEN
      qdb_utilities.syslog_warning ('SEND_EMAIL',
                                    'Unable to send email ID',
                                    email_id_in);
END send_mail;
```

I post a warning to my log, and let things go on as normal.

Which brings me to my final point for this section: if you are not already using a standard error logging API (log error, log warning, log information, trace execution etc.), please put one in place ASAP. For the Dev Gym, we use our own qdb_utilities package. But I suggest you check out the **open source logger** utility. Used by thousands around the world, very robust,

supported by some really excellent developers. No reason to build your own!

Worse-than useless re-raising of an error

OK, so you have an exception handler, you don't just let the exception escape the block. But inside that handler all you do is:

Bad Code

```
EXCEPTION
   WHEN OTHERS
   THEN
      RAISE;
END;
```

This is worse than doing nothing at all. It doesn't add value. It actually makes it harder to find the cause of the exception.

The reason is that this exception propagates to outer blocks, I sincerely hope that you do trap and log the error. At which point, the logging procedure should call the DBMS_UTILIY. FORMAT_ERROR_BACKTRACE procedure, which will give you the line number on which the exception was raised. But the back trace only traces back to the most recent raise. So, when you re-raise an exception, you lose that "original" line number.

So it's OK to re-raise an exception, just log first, so you can grab and store the back trace.

Cleaned Up

```
EXCEPTION
   WHEN OTHERS
   THEN
      logger.log_error (...);
      RAISE;
END;
```
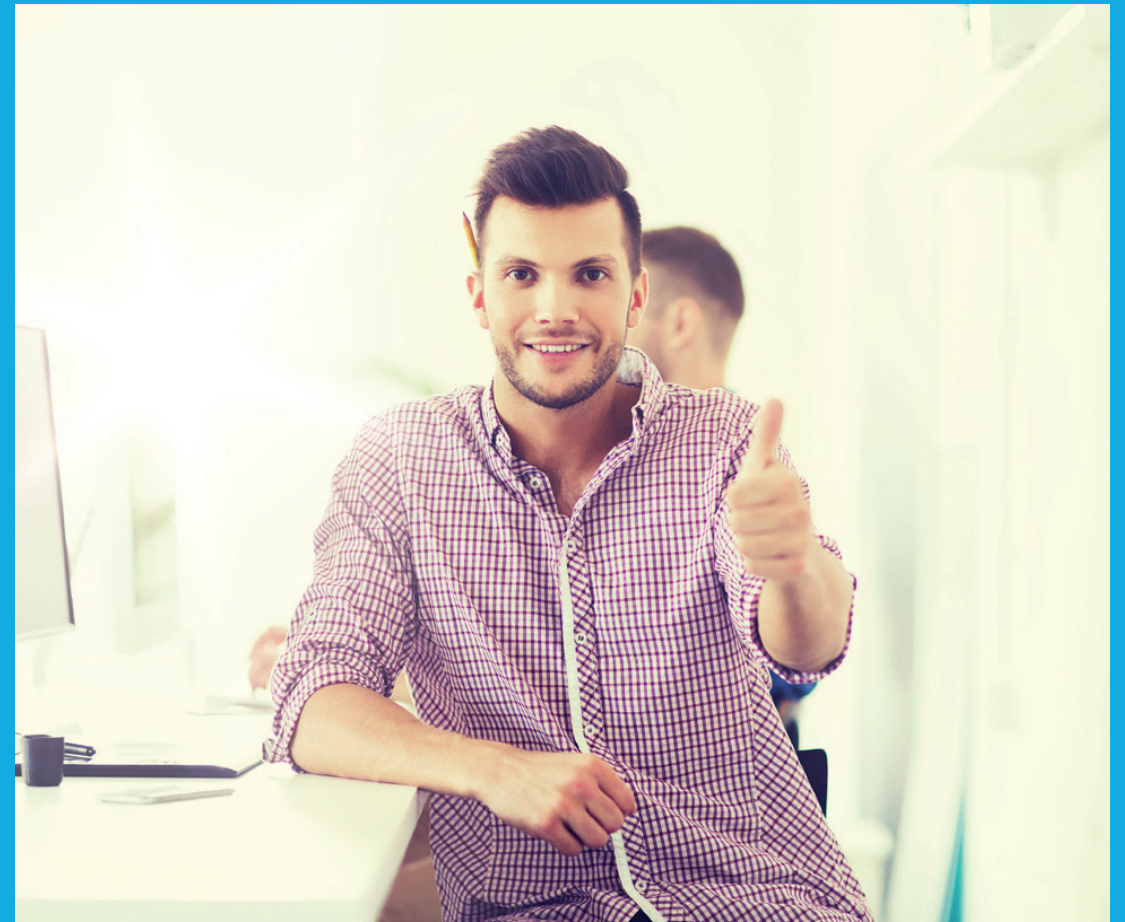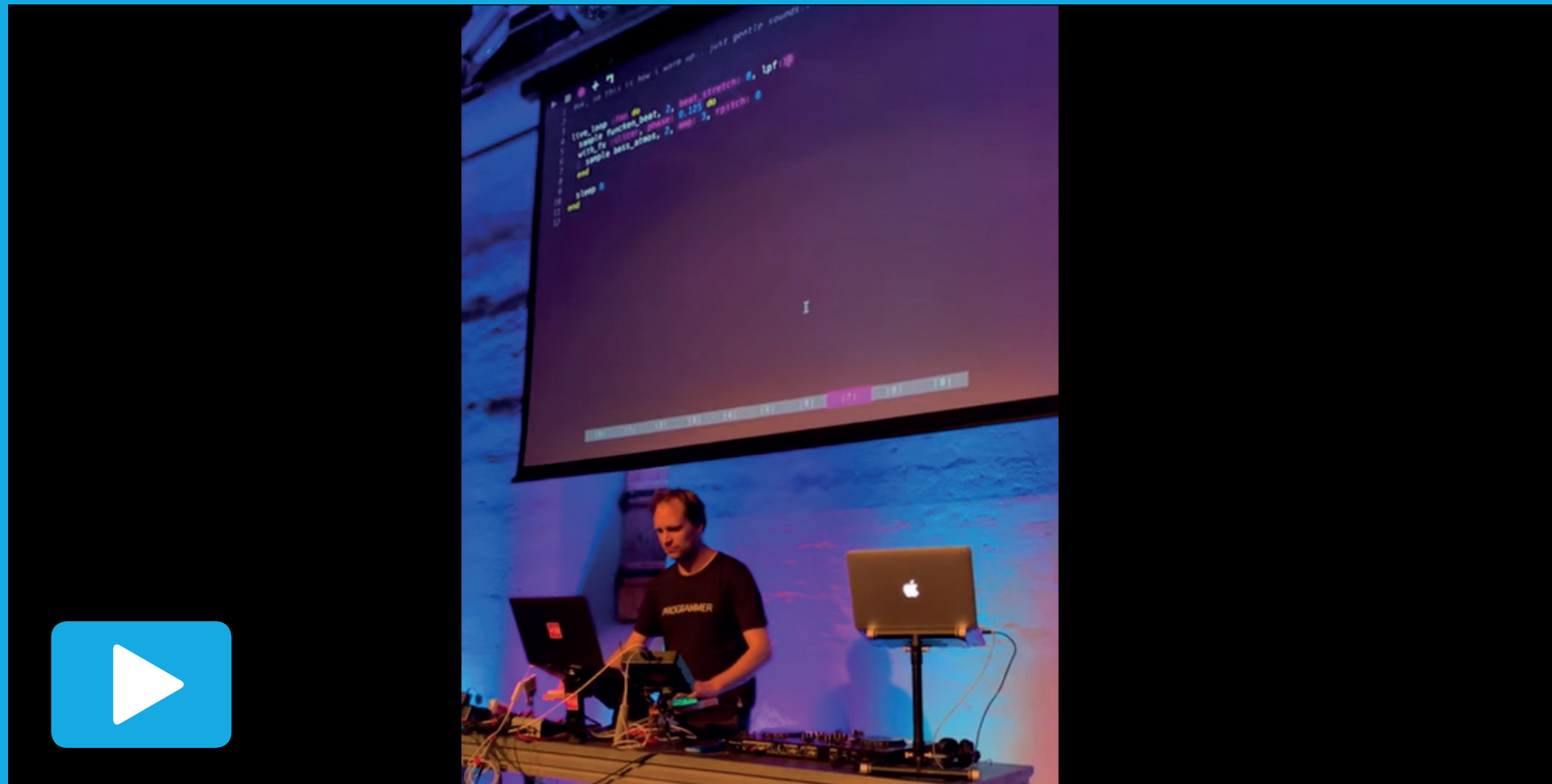
Whew

Well, I could probably go on and on, but this should be enough to give you some solid tips for writing better PL/SQL code – and inspire my readers to offer their own additions to the list.

Steven Feuerstein's Blog
The Oracle Developer Advocates
The Oracle Dev Gym - workouts, classes, and quizzes on SQL, PL/SQL and more

Techs & Nerds

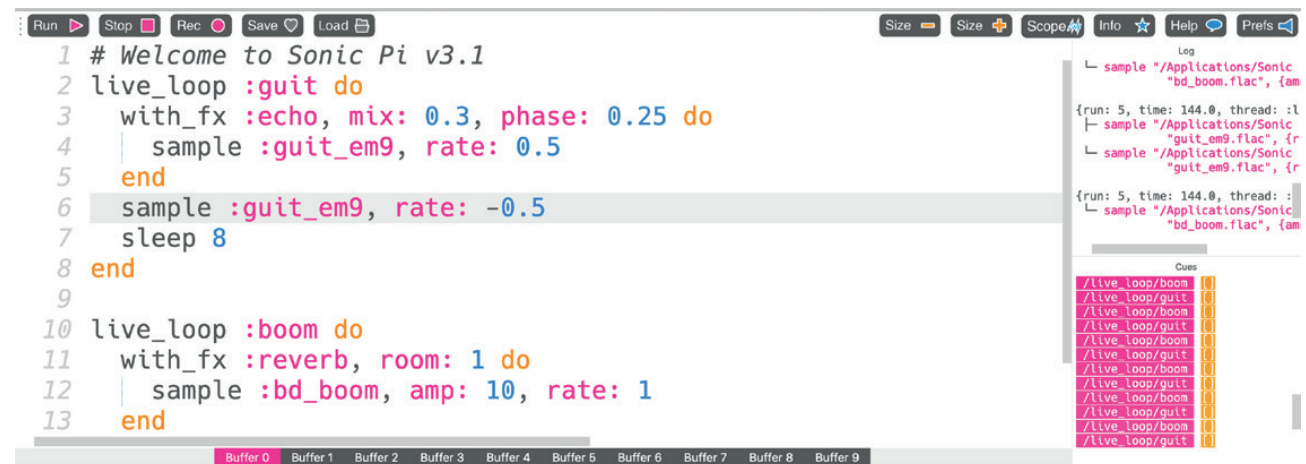# Play the Code – an Extraordinary Experience

Ann-Sofie Vikström

During the Riga DevDays from May 29 - 31, there was one speaker in particular, Sam Aaron, who left a lot of us inspired and motivated to try out for ourselves. How often does that happen?

Sam Aaron is a live coder, educator, and researcher who was the lead developer of Quil and Overtone, two Clojure-based projects for sound and image.

Sam and his team have created a domain specific, Ruby-based environment called Sonic Pi – a tool for live coding music to be used by professionals and education musicians. Basically, anyone can learn the commands little by little.

Following his keynote "Sonic Pi Livecoding", Sam gave a session in which he described the code behind his tool more deeply and then took it to the after-party. The video clip shows Sam Aaron live coding while we were dancing to the music.

Later, I downloaded **Sonic Pi** and after reading 4 steps of the instructions was up and running doing the tutorial on my Mac.

It is free, it is fun, and anyone can do it – try it yourself!

Welcome!

This is Sonic Pi
the live coding music environment

To get started please follow the
tutorial in the help system below
(which you can always access via the Help button)

and remember...
with live coding
there are no mistakes
only opportunities

Have fun and share your code
for others to jam with

40 Years of Oracle Forms

Frank Hoffmann

This year, Oracle celebrates an impressive double anniversary: 40 years of Oracle Databases and Forms. Bill Friend, Oracle's fifth member of staff after Bruce Scott, had been tasked by Larry Ellision in 1979 with developing the famous software tool. Before Christmas of the same year, Oracle released the first commercial version of Forms together with the first commercial Oracle Database and the first Reports tool two years later.

Mercedes advertised its 1979 G-Class with the claim „Stronger than time" — a slogan that could also be applied to Oracle Forms. The ensuing software evolution has already lasted 40 years. The module code from ASCII to XML. Support for Java versions from 1 to 11, database support for the versions 2.3 to 19 and Windows support from 3.1 to 2016, together with Linux, Solaris and AIX support. The tool has kept pace with these changes every stage along the way, but without users having to re-write their existing code — a huge benefit. Standard 4GL functions such as row locking, binding, and use of PL/SQL in the current database represent huge advantages over any other PL/SQL tool.

It's a fitting time to look back to the beginnings of Oracle Forms to understand the reasons why it was created. Tips from Michael Ferrante and social media led me to Bill Friend, the developer of the original version, and I had the privilege of interviewing him in detail on the subject. At 60, he is still very lively and can recall numerous details from the early days of Forms. With a 9-hour difference between our time zones, we agreed to talk in the late afternoon (CET). The story that follows and the interview are condensed from information provided by Bill Friend and an interview conducted at the end of December 2018. It was Bill's first ever interview about Oracle Forms in 40 years.

Forms sees the light of day

Forms first sees the light of day in 1979. Several interesting and exciting developments appeared in 1979. Quincy Jones gave Michael Jackson his big break with „Off the Wall". Mercedes developed the first of its G-Class series. In Japan, Namco launched „Galaxian", its first color arcade-based game, followed by the famous „Galaga" and „Pac-Man" games, long before home computers emerged as a dominant force. It was also a

year of major technical developments however, such as TCP/IP, the C programming language, and the sales launches of the Intel 8088 and Motorola 68000 processors. These developments all laid the foundations for future developments by Microsoft, Apple, and Oracle, and the spread of the Internet.

Oracle's first C tools appear

Around this time, a young developer was looking for a job. Bill Friend did what other job-seekers usually did in those days: he grabbed the Yellow Pages, flicked through to „Data Processing", and looked for suitable companies. Under „S", his eye fell on „Software Development Labs". A company with three founding members and a talented developer named Bruce Scott in their employ. Bill Friend's first phone call was answered by Bob Miner. When asked by Bill about a job, Bob replied: „We're creating the first commercial relational database management system, we're doing it on DEC PDP minicomputers in assembly language, but we want to rewrite it in C for portability. We need someone who can write some tools for the thing". Bill drove to Sand Hill Circle in Menlo Park for his interview, and was invited to lunch with Bob Miner and Larry Ellison. Once Bob had demonstrated a couple of database queries using UFI (User Friendly Interface — a forerunner of SQL Plus) and Bill had described his programming experience to Bob and Larry, the two sides quickly came to an agreement. The fact that Bill didn't have a university degree wasn't seen as a problem, nor that he had never developed anything in C at that point — that could, after all, be learned. For Larry and Bob, Bill was exactly the right person: „smart" enough for the job. Their instinct was to prove correct. Bill Friend thus became the number 5 at Oracle, the second employee recruited by the company, and the beginning of the company's expansion from 5 employees

to 500 in the years that followed. Bill immediately bought „The C Programming Language", a book by Brian Kernighan and Dennis Ritchie (K&R book), which had appeared in 1978. He had great fun teaching himself C, and had found the very job he'd always wanted.

A simple task given to Bill by Larry Ellison became Oracle Forms

Besides familiarizing himself with C, Bill Friend also got to grips with the new relational database system, working closely with Bruce Scott. Within a couple of months, he was familiar not only with C, but also with the database technology and the DB APIs.

He then had to figure out for what his tasks were going to be. In this period, no one at Oracle was supervised, much less told exactly what to do. Everyone in the company did what they did best, and with a passion. Bill however had no specific project of his own, and so he went to Larry and asked for one. Larry responded by saying that the company needed a more user-friendly way of writing data to the database than by means of an INSERT command. Bill, he suggested, could perhaps program a „prompt" to be displayed for each database column. Starting from these very vague instructions, Bill produced a detailed concept that went much further than simply satisfying Larry's request.

Bill's basic concept for Oracle Forms 1979

His intention was to make life easier for the forms developer and forms user, and to achieve a high level of productivity, in line with the 4GL theory.

4GL programming languages are designed to enable functions or complete applications to be written quickly, with the fewest possible lines of code, for a certain area of application.

These were the targets that Bill Friend had set himself:

- The design process of an input form should be interactive and easily modified.
- Users should be guided dynamically through the input form at runtime.
- The runtime engine should perform the transaction to the database dynamically. The developer of the form should be able to perform standard Insert, Update, Locking, Delete and Query transactions without having to program anything. In the event of an error, rollback should be possible. In addition, all SQL commands should be executed in the correct sequence.
- End users should be able to perform even complex queries themselves by using relational operators such as „<", „>", „LIKE" in conjunction with other variables.
- The implementation should preserve the logic of the relational model, with transactional integrity.
- User-friendly data input of relationships to other tables should also be possible by means of „lists of values".
- Pre- and post-logic triggers should be developed for automated SQL transactions, to enable the SQL logic to be extended with procedural functionality.
- The best aspects of two worlds were to be combined: Dynamically generated SQL with transaction management, and facility for use of procedural extensions with runtime triggers for application logic. (SYSTEM-R triggers were an Oracle vision and were not yet implemented in the database; they appeared first in Forms).
- The tool should be portable to all operating systems.
- A „CRT" form should make the system compatible with a range of monitors (e.g. 24x80 character grid on a cathode ray tube monitor).
- Input was also to be possible on teletypes, since not every workstation was equipped with a monitor at that time.

Using the full range of possibilities offered by C, Bill Friend developed the first version of Oracle Forms within a little over 4 weeks. Larry Ellision named the product IAF (Interactive Application Facility). The name was subsequently changed, first to „Fastforms", then to „SQL Forms", until it finally became „Oracle Forms and Reports" (from Version 3 onwards).

The birth of Oracle Reports

Larry had promised a customer a reporting tool, and Bill was asked whether he could perhaps quickly produce one. His holiday could be upgraded (to Paris or Hawaii) if he was willing to postpone it in order to complete the task. Bill took an older documentation solution (FMT) produced by Bruce Scott and added functions, similar to those in Forms. The facility to declare additional variables, select statements with variables, loops, and IF statements were also added. Around two weeks later, RPT appeared and was delivered together with the database. Bill was then able to enjoy a holiday on Hawaii with his girlfriend.

New 4GL tool produces its first success stories

IAF and RPT were a hit with Larry Ellison, and were a factor in expansion of the customer base. Everyone, from the CIA to the Bank of America, major oil and gas companies, down to small IT consulting businesses now developed „forms and reports".
At that time, customers were still tied completely to the database manufacturer and were unable to develop software applications of their own. Data input with UFI (later SQL Plus) or the new C-API was not user-friendly.

Now however, customers could both model a database of their own and create it physically with UFI, and develop user-friendly input forms and reports using IAF/RPT. Further tools developed by Bill were added in the form of EXP/IMP, which enabled databases to be exported and migrated. Bill also integrated the DATE, TIME, and TIMESTAMP data types, since time data is often very important during data acquisition.
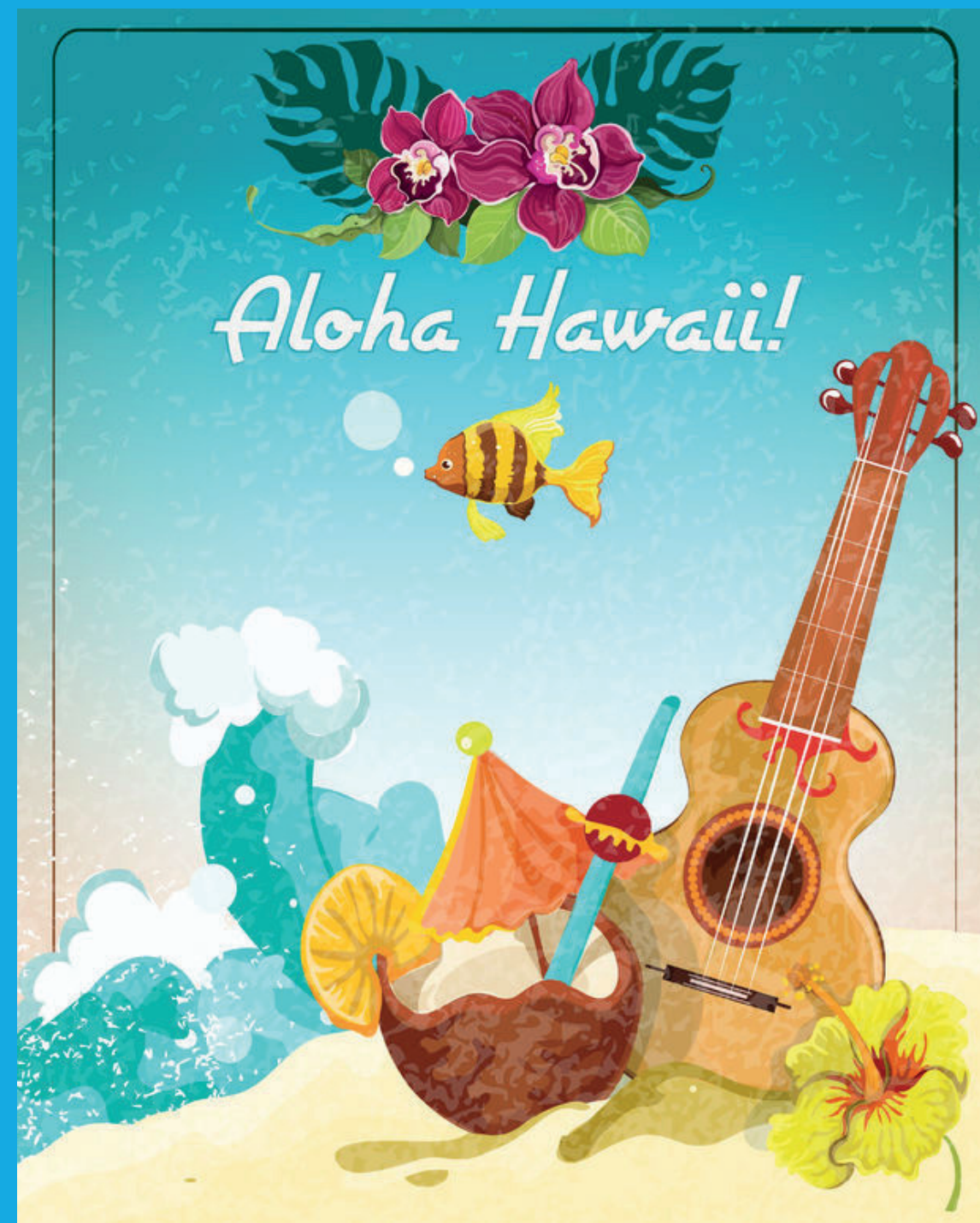
The result was a cohesive overall package. In 1985 Sohaib Abbasi became responsible for Oracle Forms. He made Forms the tool we know today. He brought PL/SQL into it long before the DATABASE and made it even more productive. Many Forms developers are still working in projects that started in 1999 with a version of his that is still in production. Forms for the Internet was designed in his time too. The idea to have a stateless client as Java Applet and a stateful server process with a solid SQLNET connection is still a powerful engine today and probably the only pure PL/SQL tool which does not need extensions. It is nearly impossible to replace Forms. For the last 15 years, it was tried many times without real success. Bill and Sohaib, who left as VP and SVP Tool division, were one main reason for Oracle's success in the first 20 years.

Forms now, in 2019

Many Forms customers have switched to the current platform (12.2.1.3), are launching „native" modernizations, and are looking ahead to the new versions, Forms 19 and Forms 20, which have already been announced. Version 19 will primarily

Sohaib Abbasi

be a maintenance release. Further changes are to be introduced with Version 20. As is now standard with Oracle Databases, the version number is based upon the release year. At present, it seems very likely that Version 19 will once again be shipped together with Reports. This will provide all customers with a little more time to evaluate alternatives. Mission times of 20 to 30 years are by no means uncommon for Oracle Forms projects.

Like Oracle itself with its Enterprise Business Suite (EBS), parts of which are still based upon Oracle Forms, customers are currently planning support and further development of their applications up until 2030. Mission times of this length are unusual for other development tools. According to Michael Ferrante, Principal Product Manager responsible for Oracle Forms, the following features are likely for the pending versions (19/20):

- REST call-up functions for external services
- Support for SSO with FSAL
- Identity Cloud Service support
- OAuth support
- UI improvements (frames, colors, custom color scheme)
- Configurable Java versions for FSAL
- Support for Java 11 FSAL (e.g. Java 11+)
- Forms Builder integration with FSAL (currently only http plugin)

Together with the changes published in Forms 12, Forms developers will have a range of features at their disposal in Forms 19 to give their forms a facelift. Forms can and should become more professional in their layout, without gray shades, with high contrast, and with color palettes from which the users benefit.

A further new option will be the outplacement of Forms development and deployment to the Oracle Cloud. In a very new development **described in a white paper in November 2018**, Mike Ferrante presents the new opportunities for Forms development (DevOps) in the cloud.

Time for Modernization or Migration?

While Oracle Forms celebrates an impressive anniversary of 40 years, companies using this technology are often told they should better replace it.

However, migrating away from Forms would be an expensive and risky journey. According to an internal Accenture study the migration of an existing Forms-based ERP system of Fraunhofer Society for Applied Science to SAP would have been 20 times more expensive than an upgrade of Forms. Migrating to other Oracle tools would be highly expensive too.

In recent years, numerous Oracle Forms customers have invested considerable sums in the software and intend to continue using it.

While Oracle has assured long-term support and reaffirmed its commitment, many customers are missing some things for Forms to be successful in the future – in the cloud and on-premises. Therefore, members of DOAG have started an initiative which includes the **Oracle Forms Modernization Petition**.

The supporters of the petition urge Oracle to bring the GUI of the runtime environment up to date and to make Forms suitable for cloud-friendly deployment (by means of JavaScript), in order to enable the developed applications to be deployed on all devices, whether desktop or mobile.

You can support the petition by taking part in the **DOAG Survey**. If you do not want to support it, you can still use the same form to tell us your opinion on Oracle Forms – let us know what you think!

Frank Hoffmann
Cologne Data GmbH
fh@cologne-data.de

Techs & Nerds

Interview with Bill Friend
(responsible for Forms from 1979 until 1985)

What are your favorite all-time features of Oracle Forms?
I would say automation of the transaction – the dynamic SQL and correct sequencing of triggers based on end-user manipulation of the form data. User guidance controlled by transactions and triggers. Productivity though (which encompasses many features) is the heart of it. That was something our competition never seemed to match. The approach was basically 4GL, and where the 4GL was no longer sufficient, you could augment the logic with procedural tools.

Oracle Forms turns 40 this year, and its popularity is growing. Did you originally foresee this success?
I'm honored that Oracle Forms, in another incarnation but true to its roots, is still in use and may be for some time to come. I never expected it to survive this long and frankly was hoping something better would have come along by

now (I had also wanted to rearchitect it to get rid of some of the limitations). There are not many examples of systems or products with that kind of lifespan in an industry that has changed so much. I think I started to feel like I had built something important and lasting in 1984, when I saw jobs being advertised for Forms and Reports developers.

Someone said once that Forms & Reports and the database were the only pieces of software that Oracle had ever made itself; the rest was sourced externally. Is that true?
At the time we were starting Oracle, Computer Associates was how Oracle is today – buying mostly everything to grow. Today, Oracle looks much more like CA in some respects. Oracle Designer, Oracle Applications (the original ones), many of the data warehouse tools including the one I built between 1998 and 2000 were done in house.

How influential was Larry Ellison in the development process of Forms over the years?
Larry is the smartest guy I ever met and also a true visionary – not just in technology, but really visionary at how to grow a company. He also has a very self-deprecating sense of humor that I found endearing and made him super-approachable. I really like the guy, but it could be frustrating at times working for him. Likewise, Bob Miner had a huge influence on me and was a great guy. The one thing everyone in the early days at Oracle had in common was that they were strong-willed, very smart, very direct, very cutting when it came to critical issues relating to the software. Everyone was absolutely committed to their own area and in continual competition with others to get the best possible result. Not just on work things – no, tennis, fingertip pull-ups in the doorways, poker, you name it. And no one was more competitive than Larry. He wanted to be the best, and for us to achieve the best results in our work. This had an influence on the products that we developed.

Bill Friend

Techs & Nerds

What is the impact of a shift from Forms, a 4GL language, to a 3GL language?

About the time I was leaving, there was a real movement towards programming forms in Java, moving away from 4GL. I'm not sure if that was a good thing. You lose a lot if your procedural (3GL) tool doesn't retain the power of the 4GL aspects: specifically, managing the transaction and all the associated SQL invisibly to the developer. All this has to be done by the developers themselves in 3GL languages. Non-procedural 4GL languages are incredibly good at expressing certain complicated things in a simple way, which is why developers love 4GL tools. But these tools lack that ability to be tuned precisely to some aspect that the 4GL was not designed to do or is outside its scope. And that's where people start hating them. Procedural languages give you the ultimate flexibility – which developers love – but then all the "free" stuff you got with the 4GL disappears. Forms, which has had automated transaction management and the facility to integrate procedural triggers since its early days, gives you that ability to mix the two paradigms.

Do you remember how big your development unit was for the initial versions of Forms?

One developer. And at the very end there was also Sohaib Abbasi, who took over after I left. He went on to become the SVP of the Tools Division. In hindsight I should have thought about hiring people to help me, but that thought never even crossed my mind as I just enjoyed doing everything myself.

In your opinion, why do we now see so little promotion of Forms by Oracle?

They are busy promoting cloud technology and they have not provided a migration path from Forms to other cloud technology or updated it sufficiently. Perhaps this is a missed opportunity. People want to push new products that they

think could replace something as old as Forms. As I said in my bio, Oracle is a very competitive place. But until and unless the new products capture the essence of what makes Forms powerful – high performance, high productivity – both for non-programmers and programmers, then Forms will likely continue, and thrive. I sometimes regret now never getting to work on a replacement myself because I have some ideas from my experience in applications and data warehousing that would have built on the core features of Forms and dealt with a lot of the limitations that it runs up against.

The interview was conducted by Frank Hoffmann on December 31, 2018. Look forward to his second interview with Sohaib Abbasi, responsible for Forms from 1985 to 2003, being featured in the next issue of ORAWORLD!

Part Cat, Part Octopus?

Lisa Damerow

GitHub is a community platform for Developers to share and discover ways to potentially simplify their daily working life. It has around 31 million members (October 2018). Every time you visit the website you will come across their "logomark" (image 1) in the upper left corner – a black and white silhouette of... well, of what exactly?

The black and white logo decorating the website is the outline of "Mona Lisa Octocat" (Mona for short), GitHub's official mascot. She is part octopus and part feline; hence how she got her name. The octopus symbolism comes from a concept called "octopus merge" that Git uses, the software that GitHub runs on. In the "octopus merge" three or more branches of development are combined. Octocat has five tentacles, one of them symbolizing a tail, whiskers as well as cat ears *(image 2)*. Her creator, graphic designer Simon Oxley initially sold the character as a clip art named "Octopuss" on iStock. GitHub, however, sought to use the image as their logo, which was not possible under the iStock license, so they bought the exclusive rights from Oxley. She even has **her own Twitter account**! In the **GitHub Octodex** – which is basically Mona's photo diary, if you will – there are hundreds of different versions of the adorable mascot in different outfits and situations.

Fun fact: Oxley also designed the original Twitter bird *(Image 3)* that was used for the very first logo in 2006. It was sold on a stock image site, so Oxley made around $ 3 off it.

| 2006 | 2009 | 2010 | 2012 |
| (Simon Oxley) | (Biz Stone + Philip Pascuzzo) | (Biz Stone + Philip Pascuzzo) | (Doug Bowman) |

Image 3: Previous Twitter logos and their creators

Sources:
https://www.quora.com/What-is-the-GitHub-logo
http://cameronmcefee.com/work/the-octocat
https://en.wikipedia.org/wiki/GitHub#Mascot

Image 1: GitHub's logomark

Image 2: Mona Lisa Octocat

Techs & Nerds

The EOUC Leaders' Summit in Riga

Heli Helskyaho

Photo credit: Vidar Eidissen

The EOUC Leaders' Summit was held during the **Riga DevDays** in Latvia. The attendees were presidents and representatives of Oracle user groups in EMEA. The main goal of these Summits is to share information with user groups and to enable collaboration. In this Summit we talked about topics like "How to keep long-lived community active and find active members?" and "Is membership of our user groups growing, shrinking, changing demographically (older, younger, gender, tech-focus) – and what can we either learn from this or change?"

We also talked about our preparations for the Oracle OpenWorld and the traditional tours that will be organized in EMEA: the **Oracle Groundbreakers EMEA Tour** and the Nordic Groundbreakers Tour through Sweden, Norway, Denmark, and Finland.

But the main concern in this meeting was that Oracle is no longer organizing these meetings; and the lack of support for EOUC collaboration is quite clear. The Summit spent quite a lot of time addressing this issue. The overall decision was that the user groups in EMEA want to continue collaborating and plan to start an umbrella organization that will facilitate the EMEA Oracle user group collaboration. The next meeting will be held in Nuremberg, Germany on November 18 at the DOAG 2019 conference. The main topic for this meeting will be the new organization.

During the meeting we also learnt about the Oracle Support Survey conducted by DOAG and the whole EOUC. Based on the survey results the meeting decided to write a letter to Oracle. In this letter the user groups represented impress their support for Oracle cloud strategy but at the same time ask Oracle to respect the fact that not all the organizations are able to go to the cloud immediately. Therefore, it would be vital for Oracle to keep supporting products also in a non-cloud environment. The

meeting also expressed the concern on regional presence for Oracle in EMEA.

I want to thank Riga DevDays for their hospitality during the Summit! The meeting was very important and could not have been organized without the support from Riga DevDays!

Users & Groups

NetSuite User Day Makes Successful Debut

Christian Luda

On May 14th the German NetSuite community gathered at the first NetSuite User Day in Berlin-Kreuzberg.

The ERP system NetSuite is enjoying increasing popularity in Germany. Almost 70 participants had come to Berlin to exchange lessons learned and best practices and to express their wishes and criticism to the NetSuite representatives present.

Contentful's premises, located in a typical Kreuzberg business park, provided the perfect ambience and underlined the community character of the fully booked event.

The lunch reception was already used for networking, before DOAG and host Contentful officially welcomed the participants and opened the lecture program. Christian Walch and Alexander Richard from the auditing and consulting firm KPMG kicked off the event. They reported on the challenges of introducing NetSuite to an international pharmaceutical company – one of the largest NetSuite projects in Europe to date.

NetSuite managers Mathias Reinecke (Regional Product Manager, Germany, Austria, and Switzerland) and James Chisham (Director of International Product for EMEA) presented the new features of NetSuite 2019.1 and – to the delight of many attendees – shared plans of a first German data center that will be available in Frankfurt in early 2020.

The event continued with exciting, practical lectures in front of a full audience. Markus Fest from Eve Systems introduced a NetSuite-based eCommerce solution, Phuong Thao Le from opinion research start-up Civey highlighted opportunities for revenue-recognition automation, and Kerstin Fischer from Alta Via Consulting presented new possibilities of parallel accounting.

At the end of the lecture program, Peter von Zimmermann provided another highlight. DOAG's topic manager for

NetSuite had previously examined the new tax administration solution SuiteTax from a German point of view and now presented useful innovations in an entertaining way, but also some weaknesses – for example, no gross amounts can be entered for travel expenses. His conclusion: Before these problems are not solved, German users should wait and see. Because a return to the classic tax administration does not exist, once you have activated SuiteTax. The NetSuite representatives took up this criticism and responded to subsequent, sometimes worried questions from the audience. James Chisham assured that there will not be a complete switch to SuiteTax until all the vulnerabilities have been resolved.

This last lecture underpinned the claim, which moderator Matthias Runte (DOAG's Deputy Topic Manager for NetSuite) emphasized once more: The NetSuite User Day should not be a promotional event, but in addition to exchange and networking also provide a place for clear criticism towards the manufacturer. A concept that matters: In the following feedback session, the wish to repeat the User Day not just once but twice a year was expressed several times.

At the final get-together in nearby Dookie, there was an animated discussion over pizza and wine and anticipation of the next event. Matthias Runte drew the following conclusion: "We managed to position ourselves as the lawyer of customers and users towards NetSuite. This is something the participants have also taken in and is certainly one of the main reasons for the invariably positive feedback, in addition to the quality of the lectures and the very successful event framework. We are well on the way to setting a very successful event format that will be very well established alongside NetSuite organized sales events."

Heli Helskyaho

Ambassador's Corner

Dear user group leaders,
I hope you and your user groups are doing well!

EOUC Leaders' meeting

The EOUC leaders' face-to-face meeting in Riga, Latvia was a great success. You can read more about it in this magazine. The next step will be planning how to start an umbrella legal entity for EOUC. This will be discussed in more details in our next meeting in Nuremberg, Germany in November. Mark your calendars: November 18th!

Oracle OpenWorld 2019

The preparations for OOW 2019 are moving smoothly. I hope to see many of you in San Francisco from September 16 - 19.

All the best to you and your user group!

Best regards,
Heli

Your Ambassadors:

If you have anything we can help with, please do not hesitate to contact us!

Ami Aharonovich
ilOUG (Israel)
ami@dbaces.com

Heli Helskyaho
OUGF (Finland)
heli@miracleoy.fi

Call for Papers

SAOUG Connect 2019

August 20, 2019
Emerald Hotel and Resort,
Vanderbijlpark, South Africa
conference@saoug.co.za

Itoug Tech Days 2020

info@itoug.it

ILOUG Techdays

Petah Tikva, Israel
https://www.iloug.org

JavaLand 2020

July 11 - August 26, 2019
Phantasialand, Brühl, Germany
https://www.javaland.eu/en/speaker/speaker/

Events

Quest Oracle Community Cloud Webinar Series

August 6 - September 10, 2019
Virtual event
https://www.questoraclecommunity.org/events/
conferences/2019-cloud-webinar-series

SOUG Day

August 27, 2019
Baden-Dättwil, Switzerland
https://soug.ch/events/soug-day_august19

POUG 2019

September 6 - 7, 2019
Wroclaw, Poland
https://www.poug.org/en/about

Oracle Open World

September 15 - 19, 2019
San Francisco, California
https://www.oracle.com/openworld

JD Edwards INFOCUS

August 20 - 22, 2019
Sheraton Denver Downtown, Denver, Colorado
http://www.questoraclecommunity.org/infocus

Nordic Oracle APEX Tour

September 3 - 6, 2019   Copenhagen, Oslo, Helsinki,
Stockholm
Denmark: https://www.doug.dk
Norway: https://ougn.no/apex-day-2019
Finland: www.meetup.com
Sweden: https://www.sweoug.se/apex2019

Quest Oracle Community Global Peoplesoft Day

September 12, 2019
Virtual event
https://questoraclecommunity.org/events/
conferences/global-peoplesoft-day

Oracle Code One

September 15 - 19, 2019
San Francisco, California
https://www.oracle.com/code-one

Events

Oracle Groundbreakers EMEA Tour 2019

October 10 - 18, 2019
Dushanbe, Istanbul, Baku, Portoroz, Rovinj, Bucharest
http://www.ogbemea.com

Oracle Groundbreakers Nordic Tour 2019

October 22 - 25, 2019
Copenhagen, Oslo, Helsinki, Stockholm
http://www.ogbnordic.com

QXW- Quest Experience Week

November 12 - 15, 2019
Virtual event
https://www.questoraclecommunity.org/qxw

Techfest 2019

December, 1 - 4, 2019
Brighton, England
https://www.ukoug.org/page/techfest19

HrOUG 2019

October 15 - 18, 2019
Rovinj, Croatia
http://2019.hroug.hr

SAOUG Connect 2019

November 10 - 12, 2019
Emerald Hotel and Resort , Vanderbijlpark, South Africa
https://www.emeraldcasino.co.za

DOAG 2019 Conference + Exhibition

November 19 - 22, 2019
Nuremberg, Germany
https://2019.doag.org/en/home

Itoug Tech Days 2020

January 29, 2020   January 31, 2020
Milan, Italy   Rome, Italy
info@itoug.it